Container Adapters Solutions

Copyright © James Raynard 2021

std::queue example

- Write a program which populates an std::queue and displays its size, front element and back element
- Modify your program to remove the front element, then display the same information
- Explain your results
  - Data in a queue is stored in order of insertion, with the oldest element at the front
  - When the front element is removed, the next oldest element becomes the new front element

Copyright © James Raynard 2021

std::priority_queue example

- Write a program which populates an std::priority_queue and displays its size and top element
- Modify your program to remove the top element, then display the same information
- Explain your results
  - The elements in a priority queue are stored in order of "importance", with the most important element at the front and the least important at the back
  - By default, the ordering is based on the element's < operator
    - For int, the elements with the highest value will be at the front of the priority queue
    - For std::string, this will be based on the alphabetical order of the elements
  - When the front element is removed, the element with the next highest priority becomes the new front element

Copyright © James Raynard 2021

std::stack example

- Write a program which populates an std::stack and displays its top element
- Modify your program to remove the top element, then display the same information
- Explain your results
  - The elements in a stack are stored in the reverse order of insertion
  - The top element will be the most recently added
  - When this is removed, the next most recently added element will become the new top element

Copyright © James Raynard 2021